NASA Contractor Report 3223

A Supersonic, Three-Dimensional Code for Flow Over Blunt Bodies - User's Manual

D. S. Chaussee and O. J. McMillan

CONTRACT NAS1-15305
JANUARY 1980

NASA

NASA Contractor Report 3223

A Supersonic, Three-Dimensional Code for Flow Over Blunt Bodies - User's Manual

D. S. Chaussee and O. J. McMillan
Nielsen Engineering & Research, Inc.
Mountain View, California

Prepared for
Langley Research Center
under Contract NAS1-15305

NASA
National Aeronautics
and Space Administration

Scientific and Technical
Information Office

1980

TABLE OF CONTENTS

1. INTRODUCTION

This report consists of a technical explanation of the code used to calculate the steady supersonic, three-dimensional, inviscid flow over blunt bodies. Because this code has been extensively documented elsewhere (e.g., refs. 1 and 2), a certain measure of brevity is possible; further details are available in these references.

In the following section (section 2), a brief discussion of the theoretical and numerical formulation of the problem is given, including exposition of the boundary and initial conditions used. In section 3, the overall flow logic of the computer program is given, in section 4, the program usage and operation are described, and in section 5, the program accuracy and limitations are discussed.

2. THEORETICAL AND NUMERICAL FORMULATION OF THE PROBLEM

2.1 Review of the Governing Equations

The fluid dynamic equations in conservation-law form governing steady, inviscid, three-dimensional, compressible flow of a nonheat-conducting gas can be written in cylindrical coordinates as follows:

$$\partial \underset{\sim}{U}/\partial z + \partial \underset{\sim}{F}/\partial r + \partial \underset{\sim}{G}/\partial \phi + \underset{\sim}{H} = 0 \qquad (1)$$

where $\tilde{\underline{U}}$, $\tilde{\underline{F}}$, $\tilde{\underline{G}}$, and $\tilde{\underline{H}}$ are four-component vectors defined as

$$\tilde{\underline{U}} = \left|\begin{matrix} \rho u \\ p + \rho u^2 \\ \rho uv \\ \rho uw \end{matrix}\right| \qquad \tilde{\underline{F}} = \left|\begin{matrix} \rho v \\ \rho uv \\ p + \rho v^2 \\ \rho vw \end{matrix}\right|$$

$$\tilde{\underline{G}} = \frac{1}{r}\left|\begin{matrix} \rho w \\ \rho uw \\ \rho vw \\ p + \rho w^2 \end{matrix}\right| \qquad \tilde{\underline{H}} = \frac{1}{r}\left|\begin{matrix} \rho v \\ \rho uv \\ \rho (v^2 - w^2) \\ 2\rho vw \end{matrix}\right|$$

Here p and ρ represent dimensional pressure and density and u, v, and w denote velocity components in the coordinate directions z, r, and ϕ. The nonlinear system (1) of four equations represents conservation of mass and momentum.

The governing set of equations is made complete by the addition of energy conservation as given by the equation for total enthalpy

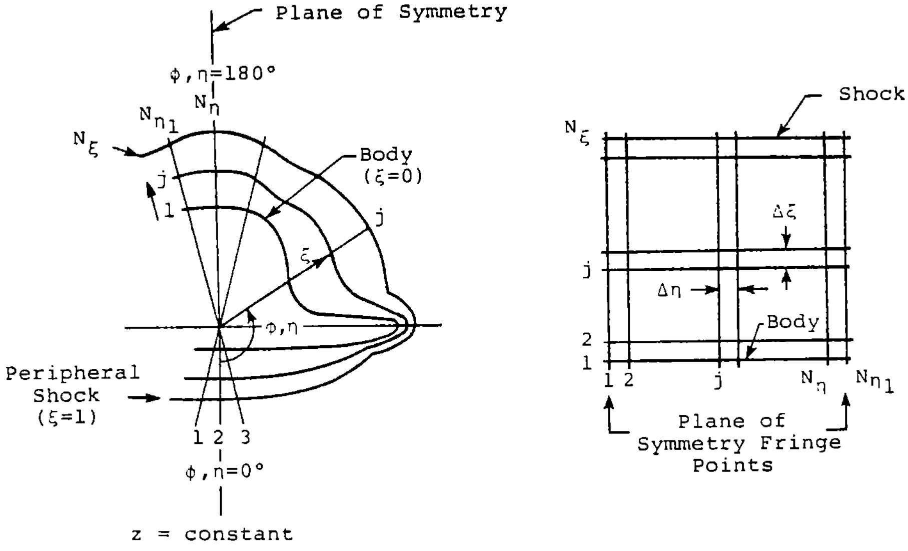
$$H_t = h(p,\rho) + q^2/2 = \text{const} \tag{2}$$

where q is the magnitude of the velocity vector and $h(p,\rho)$ is the state equation for static enthalpy. The specific formulation for h depends, in particular, on whether the gas is assumed to be perfect or everywhere in local thermodynamic equilibrium. Explicit representations for h are described later.

The vehicle geometry and the location of the outer or peripheral shock surface are represented by functions of the form

$$r_b = r_b(z,\phi), \quad r_s = r_s(z,\phi) \tag{3}$$

The function r_b is specified (sec. 2.4) and r_s is determined during the course of the numerical computation. As is common practice in problems of this type, the distance between the body and peripheral shock is normalized by a transformation of the radial variable r. This yields a rectangular computational plane whose boundaries consist of the plane of symmetry and the body and shock surfaces as shown in the sketch below.



Discretized Flow Region

Computational Plane

Mesh Description

Since the flow variables can vary rapidly in the cross flow plane, an independent variable transformation is performed in the ϕ direction to cluster points in that region of suspected large gradients. The transformation, which is also given below, was introduced by Woods (ref. 3) and has been successfully applied by Schiff (ref. 4).

The equations of the independent variable transformations are

$$z = z, \quad \xi(z,r,\phi) = (r - r_b)/(r_s - r_b), \quad \eta(\phi) = \tan^{-1}(\kappa \tan \phi) \qquad (4)$$

where κ is a free parameter with the range $0 \leq \kappa \leq 1$. The points are clustered about the wing tip region (90° plane) for small values of κ and the spacing approaches uniformity as κ approaches unity.

The change of variable procedure outlined above is applied to each vector of Eq. (1), and the resulting terms are then rearranged in conservation-law form to yield the following equation:

$$\partial\underline{U}/\partial z + \partial\underline{F}/\partial\xi + \partial\underline{G}/\partial\eta + \underline{H} = 0 \tag{5}$$

where

$$\underline{U} = \tilde{\underline{U}}$$

$$\underline{F} = \{-[r_{b_z} + \xi(r_{s_z} - r_{b_z})]\tilde{\underline{U}} + \tilde{\underline{F}} - [r_{b_\phi} + \xi(r_{s_\phi} - r_{b_\phi})]\tilde{\underline{G}}\}/(r_s - r_b)$$

$$\underline{G} = (\kappa^2\cos^2\eta + \sin^2\eta)\tilde{\underline{G}}/\kappa$$

$$\begin{aligned}\underline{H} = \tilde{\underline{H}} &+ [(r_{s_z} - r_{b_z})\tilde{\underline{U}} + (r_{s_\phi} - r_{b_\phi})\tilde{\underline{G}}]/(r_s - r_b) \\ &- (1 - \kappa^2)\sin(2\eta)\tilde{\underline{G}}/\kappa\end{aligned}$$

The finite-difference analogue of Eq. (5) is integrated with respect to the hyperbolic coordinate z to yield values of the conservative variable $\underline{U}$. Subsequent to each integration step, the physical flow variables p, ρ, u, v, and w must be decoded from the components u_i of $\underline{U}$. This necessitates the solution of five simultaneous nonlinear equations consisting of Eq. (2) together with the four elements u_i. The velocity components v and w are easily found and are given by

$$v = u_3/u_1, \quad w = u_4/u_1 \tag{6}$$

If the u_i along with Eqs. (6) are used to eliminate the explicit dependence of p, ρ, v, and w from Eq. (2), one obtains the following implicit expression for the velocity, u:

$$E(u) = u^2/2 + h[p(u),\rho(u)] - \Gamma/2 = 0 \tag{7}$$

where

$$p(u) = u_2 - u_1 u, \ \rho(u) = u_1/u, \text{ and } \Gamma = 2H_t - [u_3^2 + u_4^2]/u_1^2 \tag{8}$$

The decoding procedure is now reduced to a problem of root finding, i.e., the z-velocity component u that satisfies Eq. (7). Two roots exist; one corresponds to subsonic flow and is discarded since in the results presented u is always supersonic, and the other corresponds to supersonic flow and gives the desired solution. The procedure for solving Eq. (7) depends on whether a perfect or real gas is being considered and consequently on the function $h(p,\rho)$.

For a perfect gas $h(p,\rho)$ is simply related to pressure and density and when combined with Eqs. (7 and 8) yields a quadratic equation that can be solved to find an analytical representation (ref. 5) for the supersonic velocity u.

For a real gas no such simple explicit functional relationship exists. The conventional procedure (refs. 6 and 7) for evaluating real gas state relations is to use a combination table lookup and curve-fitting procedure. Such a scheme is adopted here. A particularly rapid Fortran-language computer code called RGAS is available that returns values of static enthalpy h, speed of sound, temperature, and entropy, with either pressure and density or pressure and entropy as independent variables.

To find the roots of Eq. (7) for a real gas, a root finding algorithm is employed. The "successive linear interpolation" scheme described by Dekker (ref. 8) was found to be particularly efficient. Slightly more than seven iterations are required, on the average, to find the desired supersonic root for a given U (three iterations is the happenstance absolute minimum).

Three-dimensional numerical integration methods (see, e.g., refs. 6, 7, and 9) which employ physical rather than conservative dependent variables are generally able to include state relations in a more direct manner since a decoding procedure is not necessary. Only half again more computational time than a perfect-gas calculation is required for those codes compared to a factor of about four when conservative variables are employed. However, the ability to capture shocks by use of conservative variables is worth the additional cost.

2.2 Boundary Conditions

In this section the boundary condition schemes applied at the body and shock surfaces are discussed briefly. At the body the surface tangency condition

$$\underline{q} \cdot \underline{n}_b = 0 \tag{9}$$

is imposed where $\underline{q}$ is the velocity vector

$$\underline{q} = u\underline{i}_z + v\underline{i}_r + w\underline{i}_\phi$$

and $\underline{n}_b$ is the outward unit normal to the body. The second-order predictor-corrector scheme, MacCormick's predictor and Eq. (15) below, is first applied at the body to yield the conservative variables. These variables are then decoded [see discussion following Eq. (6)] to obtain $\bar{p}^{n+1}_{1,j}$, $\bar{\rho}^{n+1}_{1,j}$, $\bar{u}^{n+1}_{1,j}$, $\bar{v}^{n+1}_{1,j}$, and $\bar{w}^{n+1}_{1,j}$ at z^{n+1}. In general the resulting velocity vector $\bar{\underline{q}}^{n+1}_{1,j}$ based on the predicted velocity components will not satisfy the surface tangency condition Eq. (9) and, in fact, will be rotated out of the surface tangent plane by a small angle $\Delta\theta$. In applying Abbett's method (ref. 10) this angle can be determined from the following equation:

$$\Delta\theta = \sin^{-1}\left(\bar{\underline{q}}^{n+1}_{1,j} \cdot \underline{n}_b / \bar{q}^{n+1}_{1,j}\right) \tag{10}$$

where $\bar{q}^{n+1}_{1,j}$ is defined to be the magnitude of $\underline{\bar{q}}^{n+1}_{1,j}$ and the outward unit normal $\underline{n}_b$ to the body can be calculated from (see ref. 5)

$$\underline{n}_b = \frac{\nabla f_b}{|\nabla f_b|} = \frac{-r_{b_z}\underline{i}_z + \underline{i}_r - (r_{b_\phi}/r_b)\underline{i}_\phi}{\left[r^2_{b_z} + 1 + (r_{b_\phi}/r_b)^2\right]^{1/2}} \tag{11}$$

The function $f_b = r_b - r_b(z,\phi) = 0$ describes the body surface [Eq. (3)]. Once $\Delta\phi$ is determined the velocity vector is rotated back into the surface tangent plane by imposing a simple compression or expansion wave.

If $\Delta\phi$ is positive, then an expansion is necessary for the rotation of $\underline{\bar{q}}^{n+1}_{1,j}$ and if $\Delta\theta$ is negative a compression wave is required. The corrected value of the static pressure is found from the integral relation (ref. 12) for the Prandtl-Meyer turning angle $\nu(p;H_t,s)$ which depends on pressure and has the total enthalpy and entropy as parameters. The corrected value of pressure is found by solving

$$\nu(p^{n+1}_{1,j};\ H_t,s) = \nu(\bar{p}^{n+1}_{1,j};\ H_t,s) + \Delta\theta \tag{12}$$

for the pressure $p^{n+1}_{1,j}$. In this equation $\Delta\theta$ is given by Eq. (10). If $\Delta\theta$ is sufficiently small, Eq. (12) can be inverted and solved analytically for $p^{n+1}_{1,j}$ only in the case of a perfect gas (ref. 5).

For a real gas, Eq. (12) can be inverted by the use of a table-lookup method. The isentropic flow assumption underlying Abbett's boundary condition procedure requires that the table be generated only once at the very beginning of a flow field calculation when the entropy on the body stream surface is known. The table elements are pressure and Prandtl-Meyer turning angle ν. The procedure for generating the table is described by Hayes and Probstein (ref. 12).

The pressure $p_{1,j}^{n+1}$ appearing in Eq. (12) is determined by first finding $\nu(\bar{p}_{1,j}^{n+1}; H_t, s)$ from the table with the predicted pressure $\bar{p}_{1,j}^{n+1}$ as the argument. The angle $\Delta\theta$ given by Eq. (10) is then added to the result and the desired value for the corrected pressure $p_{1,j}^{n+1}$ is then found from the same table with $\nu(\bar{p}_{1,j}^{n+1}; H_t, s) + \Delta\theta$ as the argument. The remaining flow variables $\rho_{1,j}^{n+1}$, $u_{1,j}^{n+1}$, $v_{1,j}^{n+1}$, and $w_{1,j}^{n+1}$ are then obtained in the same fashion as described in reference 5.

The outer boundary of the computational plane is the peripheral shock wave which completely encompasses the disturbed flow region. In this report a procedure is used whereby the peripheral shock wave (i.e., the shock for which freestream conditions are maintained on the upstream side) is treated as a discontinuity. The Rankine-Hugoniot relations are satisfied exactly across this discontinuity.

There are many ways of incorporating a sharp-shock calculation in a numerical algorithm, including method of characteristics and simpler methods such as those of Kentzer (ref. 11), Barnwell (ref. 13), and Thomas, et al. (ref. 14). Thomas' scheme is used here since it is relatively simple to implement and, furthermore, comparisons with other methods, including a full method of characteristics, have shown it to yield sufficiently accurate results for our purpose.

The pressure downstream of the shock wave is the basic variable on which all other shock-wave variables depend. Its estimated value $p_{N_\xi,j}^{(1)}$, $2 \leqq j \leqq N_n$ at $(n + 1)\Delta z$, is first found by the predictor step [Eq. (16a) below]. Since all other variables associated with the shock wave can be expressed as functions of pressure through the Rankine-Hugoniot equations, their values can then be found. The pressure $p_{N_\xi,j}^{n+1}$ is then recomputed by the corrector, (Eq. (16b) below), and the dependent flow variables are adjusted accordingly.

The boundary conditions at the planes of symmetry are applied in the conventional manner whereby the fringe points (see previous sketch) are filled with data reflected across the planes of symmetry, e.g., $p_{i,1} = p_{i,3}$, $w_{i,1} = - w_{i,3}$, etc.

2.3 Initial Conditions

It is necessary to have an axis-normal starting plane to begin the calculation. All variables, the shock location and the shock slopes must be specified on this plane. There are two ways of obtaining this starting solution. The first is using the pointed-cone starting solution procedure which is contained internally within this code. The second is for blunt bodies; here, the starting-plane solution is obtained from a separate computer code such as Barnwell (ref. 13) or Moretti (ref. 15).

2.4 Finite-Difference Scheme

Since only the peripheral shock is treated as a sharp discontinuity and the others are "captured" by the difference algorithm, of prime importance is the selection of the finite-difference scheme to be used to advance the field points, i.e., the points for which $2 \leqq i \leqq N_\xi - 1$ and $2 \leqq j \leqq N_\eta$ in the sketch above. It has been found (ref. 17) that MacCormack's scheme (ref. 16), which is an accurate predictor-corrector scheme, is the most efficient second-order algorithm to use in a shock-capturing technique.

The algorithm can be written as

$$\underline{U}^{(1)}_{i,j} = \underline{U}^{n}_{i,j} - \frac{\Delta z}{\Delta \xi}\left(\underline{F}^{n}_{i+1,j} - \underline{F}^{n}_{i,j}\right) - \frac{\Delta z}{\Delta \eta}\left(\underline{G}^{n}_{i,j+1} - \underline{G}^{n}_{i,j}\right) - \Delta z \underline{H}^{n}_{i,j} \qquad (13)$$

$$\underline{U}_{i,j}^{n+1} = \frac{1}{2}\left[\underline{U}_{i,j}^{n} + \underline{U}_{i,j}^{n(1)} - \frac{\Delta z}{\Delta \xi}\left(\underline{F}_{i,j}^{(1)} - \underline{F}_{i-1,j}^{(1)}\right) - \frac{\Delta z}{\Delta \eta}\left(\underline{G}_{i,j}^{(1)} - \underline{G}_{i,j-1}^{(1)}\right) - \Delta z \underline{H}_{i,j}^{(1)}\right] \tag{14}$$

where

$$\underline{U}_{i,j}^{n} = \underline{U}(n\Delta z, i\Delta\xi, j\Delta\eta)$$

$$\underline{F}_{i,j}^{n} = \underline{F}(\underline{U}_{i,j}^{n}, n\Delta z, i\Delta\xi, j\Delta\eta)$$

$$\underline{F}_{i,j}^{(1)} = \underline{F}\left[\underline{U}_{i,j}^{(1)}, (n+1)\Delta z, i\Delta\xi, j\Delta\eta\right.$$

At the body ($i = 1, 2 \leq j \leq N_\eta$) Abbett's (ref. 10) scheme is used to satisfy the surface tangency condition as is discussed above. It relies on information provided by the finite-difference scheme. The numerical algorithm used for the field points, however, cannot be used on the surface since it requires points on either side of the point being advanced and thus data at a set of points that would lie within the body. Consequently, a special second-order accurate algorithm was constructed which requires data only on or outside the body. This scheme uses the predictor step of MacCormack's method (ref. 16), Eq. (13) followed by the corrector step given by

$$\underline{U}_{i,j}^{n+1} = \frac{1}{2}\left[\underline{U}_{i,j}^{n} + \underline{U}_{i,j}^{(1)} - \frac{\Delta z}{\Delta \xi}\left(\underline{F}_{i+1,j}^{(1)} - \underline{F}_{i,j}^{(1)}\right) - \frac{\Delta z}{\Delta \eta}\left(\underline{G}_{i,j}^{(1)} - \underline{G}_{i,j-1}^{(1)}\right) - \Delta z \underline{H}_{i,j}^{(1)} + \frac{\Delta z}{\Delta \xi}\left(\underline{F}_{i+2,j}^{n} - 2\underline{F}_{i+1,j}^{n} + \underline{F}_{i,j}^{n}\right)\right] \tag{15}$$

where i = 1. After MacCormack's predictor and Eq. (15) have been used to advance the data at the body, then Abbett's scheme is used as a final corrector.

At the shock wave, a predictor-corrector sequence is again used and requires data at the shock and one point below it. The algorithm is as follows:

$$\underline{U}^{(1)}_{i,j} = \underline{U}^{n}_{i,j} - \frac{\Delta z}{\Delta \xi}\left(\underline{F}^{n}_{i,j} - \underline{F}^{n}_{i-1,j}\right) - \frac{\Delta z}{\Delta \eta}\left(\underline{G}^{n}_{i,j+1} - \underline{G}^{n}_{i,j}\right) - \Delta z \underline{H}^{n}_{i,j} \qquad (16a)$$

$$\underline{U}^{n+1}_{i,j} = \frac{1}{2}\left[\underline{U}^{n}_{i,j} + \underline{U}^{(1)}_{i,j} - \frac{\Delta z}{\Delta \xi}\left(\underline{F}^{(1)}_{i,j} - \underline{F}^{(1)}_{i-1,j}\right) - \frac{\Delta z}{\Delta \eta}\left(\underline{G}^{(1)}_{i,j} - \underline{G}^{(1)}_{i,j-1}\right) - \Delta z \underline{H}^{(1)}_{i,j}\right] \qquad (16b)$$

where $i = N_{\xi}$. Equations (16a) and (16b) are used in conjunction with the Rankine-Hugoniot relations as described above, to determine the peripheral shock slope.

The integration stepsize is based on the analysis presented in reference 5, in which formulas for the projections of the slopes of the characteristics in the z-ξ and z-ϕ planes are given and, with a slight modification due to the clustering transformation of ϕ, can be used to determine the maximum possible stepsize.

3. OVERALL PROGRAM FLOW LOGIC

The overall logical flow of the program is described in this section. PROGRAM MAIN is the executive program for this computer code. It controls the main flow of the program logic and it starts by calling SUB.GEOM3 which reads the geometry description. Next SUB.INPUT is called from which the flow conditions and the operating controls are read. Program MAIN then calls SUB.INITA. This initializes the starting plane and defines various physical constants. To complete the initialization process SUB.BNDRY(2) is called to fill in the data at the planes of reflection. This completes the process of getting the code ready to march downstream from an axis normal initial data plane.

SUB.EIGEN is called to calculate the initial marching stepsize based on the grid size and the values of the flow variables. The program then prints out the entire flow field and outer shock structure. This is done by calling SUB.OUTPUT.

The marching process is done by looping from 1 to NITER and is controlled by calling two subroutines, EIGEN and DIFFR. The stepsize of the marching process is determined every ICONST(49) iterations by calling SUB.EIGEN. Once the stepsize is determined SUB.DIFFR is called to perform MacCormack's prediction-correction sequence. Within SUB.DIFFR flow field variables on the boundaries are calculated by SUB.BNDRY and SUB.SHOCK.

After completing the iteration loop, SUB.OUTPUT is called to furnish a final solution.

4. PROGRAM USAGE AND OPERATION

The input data options for the code are described below in terms of the card number, its format, and the variables which are defined by that card. All geometry inside the program is expressed in terms of body-oriented coordinates, z (longitudinal), r (radial, normal to z), and ϕ (radial angle around the z-axis counterclockwise, starting at zero radians straight down), or functions and transforms thereof. The unit for z, r and all lengths input are arbitrary, a user's choice, requiring only that they be consistent throughout each case. In the case of angular dimensions, it has been found more convenient to input them in degrees and let the program convert and store them in radians.

Card No.	Format			Variables
1	8I5	NSEG		# of Segment Points
		KIND		Flag for Kind of Segment 0 = Sphere or Circulr Ogive 1 = Circular Cone 2 = Circular Cone with Flat Cut
2	8F10.6	ZSEG		Z - Station Initiating Segment
3	8F10.6	RSEG		r - Coord. Initiating Segment
4	8F10.6	DSEG		Distance from Centerline to Flat Chord, Initiating Segment
5	8F10.6	ASEG	ϕ_{seg}	Angle Between Straight Down and DSEG Initiating Segment
5a	3F10.6	ZC		Z at Center of Longitudinal Arc
		RC		r at Center of Longitudinal Arc
		RADIUS		Radius of Longitudinal Arc

Card 1 (Format 8I5) specifies NSEG (=number of segment points) in its first field, and KIND flags in the next NSEG fields (up to 7 maximum). The segment points, whose dimensional specifications are punched on cards 2 through 5, bracket NSEG-1 segments. The last point usually initiates an infinite extension of the last bracketed segment.

The KIND flags tell what kind of contour each segment point initiates, where 0 = Circular Arc with Circular Cross Section, 1 = Straight Line with Circular Cross Section (Forecone, Cylinder, or Aftcone), and 2 = Cut Cone (Circular Cross Section Truncated by a Flat Cut). It should be noted that KIND = 0 is not restricted to spheres, but can also be used for circular arc ogives of circular cross-section. It has been used, for example, in rounding off contours approaching a planar axis-normal base. In such a case the KIND = 0 flag initiating the last segment is also the last 0 flag, to avoid calling an extra card later. A sphere-cone-arc would have NSEG = 4 and KIND = 0,1,0,1.

The next four cards contain data describing successive cross sections (up to 7) in successive fields of each card, using Format 7F10 (with the punched decimal points governing the decimal point location in each field). With the body segmented into up to six physical contours of types KIND, each bounded by a pair of z-planes, the z-plane coordinates called ZSEG are specified on card 2. The radius of each cross-section, RSEG, is specified in corresponding card fields on card 3. If a planar surface is to truncate any circular cross section, the radial distance from the centerline to the midpoint of the chord formed is DSEG on card 4, for those fields where the planar surface is on the vehicle.

The orientation of a planar surface cut is not limited to the case of $\phi = 0$. Card 5 provides specification of ASEG, the angle ϕ in degrees subtended by DSEG in each z-plane of applicability. To define a plane, ASEG should be constant.

Card 5 contains circular-arc data, and is read only if a KIND = 0 flag appears within the fields activated by NSEG on card 1. More than one longitudinal circular arc may be specified, and one card must be read for each KIND = 0 segment initiated on card 1. The three data items of card 5a are ZC (Longitudinal Location of Circular-Arc Center), RC (Radial Location of Circular-Arc Center), and RADIUS (Longitudinal-Circular-Arc Radius). It is important that the data of cards 2 through 5a be accurate to at least 5 significant figures in order that successive contours meet smoothly at points of tangency such as sphere-cone or cone-arc junctures.

It should be noted that geometry specifications usually should encompass all regions of the configuration bracketed by the z value at the input plane and the final ZMAX specified on a later control card. However, the last ZSEG specified may be less than ZMAX if the kind of contour beyond the last ZSEG continues as a qualitatively unchanged extension of the NSEG-1 contour bracketed by the last ZSEG. On the other hand, if the input plane z-station is on a cone downstream of a sphere-cone juncture or is exactly at such a juncture, the first geometric contour described may be the cone (if the user so desires) and not the sphere, and that cone may be described by use of its apex rather than by the sphere-cone juncture as the first ZSEG. Finally, if a configuration is so variegated as to consist of more than six different segments (such as sphere-cone-cylinder-flare-cone-cylinder-aftcone-arc), such a case can be run by punching out a restart plane near the end of the sixth segment and starting a new case with the sixth segment geometry respecified as the first of the aft-region series.

Only eight basic cards are needed to input ordinary case data and controls. There are several options which also exist that either require additional cards or not as many cards. The ordinary cards and some options will be described following the glossary of the input cards.

Card No.	Format		Variables
(Cards 6-12 are read in SUB.INPUT)			
6	3E15.6,5X,I5	XMACH	Mach number
		ALPHA	angle of attack (degrees)
		GAMMA	ratio of specific heats
		NREAL	0 for perfect gas, -1 for real gas (pointed cone starting solutions are generated internally for perfect gas option only)
7	3F10.5	PHIFD	meridional angle about which points are clustered
		RK	meridional clustering parameter (0 for no clustering)
		RJ	radial clustering parameter (0 for no clustering)
8	5I5	NIT	No. of points between body and shock (max = 20)
		NIPHI	No. of intervals in meridional direction (max = 36)
		NITER	No. of integration steps desired (when ZEND is specified set NITER to 99999)

		ICONST(49)	Stepsize is computed every ICONST(49) iterations (5 is nominal)	
		NCONE	1 for pointed cone solutions, 2 for all other geometries	
9	3F10.5	CONST(9)	Courant No. (usually 0.9)	
		CONST(4)	Radial dissipation constant	
		CONST(5)	Meridional dissipation constant	
10	5I5	DISK 1	1 reads solution from tape, 2 writes solution on tape, 3 does nothing (logical unit 12)	
		DISK 2	1 reads solution from tape, 2 writes solution on tape, 3 does nothing (logical unit 11)	
		TAPE 1	1 does nothing, 2 stores body shape and writes data on tape each Z station, 3 writes data only (logical unit 9)	
		TAPE 2	1 does nothing, 2 reads starting solution from punched cards, 3 stores solution on punched cards when exiting (logical unit 7). If TAPE2 = 1 and DISK 1 and DISK 2 = 2 or 3 a pointed cone solution will be generated for the perfect gas case only	
		NTDSOS	0	
11	2F10.5,3I5	ZBS	increment in z for printing shock and body variables (ZBS > ZEND if not desired)	Print Based on Z Stations
		ZFLD	increment in z for printing field variables (ZFLD > ZEND if not desired)	
		ITPRTB	No. of iterations for printing shock and body variables (ITPRTB > NITER if not desired)	Print Based on Number of Iterations
		ITPRTF	No. of iterations for printing field variables (ITPRTF > NITER if not desired)	
		NCASE	If > 0, new case follows	

(The following card contains values used in force and moment calculations or in shifting the origin of the pointed cone starting solution.)

12	5F10.5,I5	DIAM	length used in calculating reference area; usually maximum diameter
		ALENGT	reference length used in calculating moments
		ZREF	moment reference center
		ZCG	center of gravity location for static margin calculation
		ZSHIFT	the value of Z which corresponds to the starting cone origin; if no shift set = 0
		IFANDM	0, force and moment calculation 1, no force and moment calculation

(If starting solution is to be read from punched cards (TAPE 2 = 2), the following three cards are read in main program. If solution is read from magnetic storage device, these are not required.)

12a	5E15.6	XMACH, ALPHA, GAMMA, RK, PHIFD (defined above)		
12b	5E15.6	RJ (defined above)		
12c	3I5,4E15.6	NIT, NIPHI, NREAL, (defined above)		
		P1INF	free stream pressure,	Real Gas Option Only (Dimensional)
		R1INF	free stream density,	
		V1INF	free stream velocity,	
		GASCON	gas constant (1716.0 for air)	

[If NREAL = -1, gas tables are place here and will be read in SUB.RGAS (523 cards for equilibrium air)]

(If TAPE 2 = 2 punch card starting solution is placed here. The first card is the Z station of the starting plane and is followed by flow variables at each node.)

Card 6 (FORMAT 3E15.6,5X,I5) contains the basic free-stream flow velocity, angle of attack, gamma, and the operating control for the type of gas being employed. For ideal gas cases, GAMMA (ratio of specific heats) is held constant throughout the flow field. The free-stream Mach number, XMACH, is also a key parameter in the calculation of dimensionless properties throughout the flow field. It should be noted that the angle of attack, ALPHA, is to be input in degrees. NREAL is the flag specifying the gas type, 0 for ideal gas and -1 for equilibrium real gas. Pointed-cone starting solutions are generated for the ideal gas case (NREAL=0) only.

Cards 7 through 12 contain the operating controls of the program. Card 7 determines the meridional location and the amount of clustering both meridionally and radially. The default values are zero(0). Card 8 contains the grid size, NIT and NIPHI, the maximum number of iterations, NITER, the increment between stepsize determinations ICONST(49), and the body-surface boundary condition, NCONE. CONST(9) is a Courant-number factor governing the ratio of stepsize actually used to that allowable by stability criteria and it is found on Card 9. Also, the fourth-order smoothing parameters, CONST(4), and CONST(5) are defined on this card. If needed, these parameters are typicaly on the order of 0.1. Card 10 contains the parameters that are used for storing or reading a solution from a peripheral device such as a tape or punched cards. Card 11 determines the print controls for this program and NCASE which is set to a value greater than zero if cases are to be run in succession. Card 12 defines various force and moment parameters and the value of z which corresponds to the starting-cone origin. The default for ZSHIFT is 0.0.

Cards 12a, b, and c are read only if a starting solution is read from punched cards (TAPE=2 on card 10); these are read in the main program.

The gas tables for equilibrium air are read in SUB.RGAS if NREAL = -1 in this location, followed by a punched card starting solution if TAPE2=2. The first card is the z-station of the starting plane.

Next are cards 13 and 14 which are used to change the preceding program control variables at a specified z station and to initialize the forces and moments. If no altering is desired, one card is required and the z-station should be set to a value which is greater than ZEND. Card 14 is necessary only if IFANDM = 0 and NCONE = 2 and precedes the first ZALTER card (Card 13). Any other ZALTER cards (Card 13) are placed after card 14.

(The following card(s) is used to change the program control variables at preselected longitudinal (Z) stations and is read in Program MAIN. At least one card is required if no modifications are asked for. In this case ZALTER should then be > ZEND.)

Card No.	Format	Variables	
13	F10.5,I2,I3, 6F10.5,I2,I3	ZALTER	Z station where altering occurs
		NITA	new NIT
		NIPHEA	new NIPHI
		RJA	new RJ
		RKA	new RK
		PHFDA	new PHFD
		STP	0, stepsize determined automatically >0, value of desired constant stepsize
		DISS1	new CONST(4)
		DISS2	new CONST(5)
		NSWCH1	0, new MacCormack 1, old MacCormack
		NSWCH5	0, no entropy relaxation 1, entropy relaxation

(The following card is used to initialize the force and moment calculations, and is read in SUB.COMPUT. This card is needed only if IFANDM = 0. *If NCONE = 2 and IFANDM = 0 this card is read before the first card 13 otherwise it is read after all card 13's.)

14	6F12.8	FTX FTY FTZ	initial plane forces in the z, r, φ direction
		RMTX RMTY RMTZ	initial plane moments in the z, r, φ direction

There are error messages which are printed to aid the user when problems occur. The messages, locations, and their explanations follow below.

SUB.BNDRY

In this routine, there are three error checks which are denoted by ICHECK. If ICHECK = 1, the pressure is negative, ICHECK = 2 the density is negative and if ICHECK = 3, the local Mach number squared minus 1 is less than zero $[(M_L^2 - 1) < 0)]$. These errors are sometimes due to erroneous body shape.

For each ICHECK a sequence of two error messages are printed out. One before the correction and one after. These are as follows:

```
ERROR CHECK-NEGATIVE PRESSURE IN BNDRY
 K = _____ Z = _____
 P = _____ RHO = _____ U = _____ V = _____ W = _____ ICHECK = _____

MODIFICATION INSTITUTED
 P = _____ RHO = _____ U = _____ V = _____ W = _____
```

SUB.PMYTURN

In this routine, if the Prandtl-Meyer turning angle is too great for the flow conditions, the variables are set to values which correspond to a final turning Mach number of 100. This usually occurs due to a rapid change of the body shape.

```
----BODY TURN STOPPED AT M2 = 100.0----
```

SUB.EIGEN

There are three error messages in this routine. The first is for a negative sound speed which can occur for either a negative pressure or density value. The second and third messages usually occur in the eigenvalue calculations. These messages can occur due to the geometry of the vehicle or for a local axial Mach number less than one.

```
ERROR CHECK -SPEED OF SOUND IN EIGEN. J = ____ K =

ERROR CHECK -SIGMA-BAR-1 IN EIGEN. J = ____ K =

ERROR CHECK -SIGMA-BAR-2 IN EIGEN. J = ____ K =
```

This concludes the explanation of the major error messages.

5. PROGRAM ACCURACY AND LIMITATIONS

In general, this program can be used for inviscid, supersonic/hypersonic flow past bodies with a bisymmetry plane (no yaw) at angles of attack. Conservative estimates of applicable ranges of Mach number and angle of attack are $M_\infty \geq 2$ and $\alpha \leq 25$.

When using the geometry subroutine that is included, one is restricted to non-winged bodies. This can easily be rectified by changing the geometry package (SUB.GEOM3). The necessary information for the computer code is the body radius as a function of axial (z) location and meridional (ϕ) location. This is furnished to the computer code as the radius of the body (RB), the axial slope of the body radius (R_{B_z}) and the meridional slope of the body radius (R_{B_ϕ}).

Numerically, the accuracy is second-order in time and space but to determine the actual physical accuracy it is best to compare with experimental results. These comparisons then guide the user in his descision as to how many grid points are needed to satisfactorily resolve the flow field. Note that if the first radial mesh spacing is a fairly large proportion of the body radius, erroneous solutions can occur.

The main limitations of the current version of the program are:

1. The flow in axial direction must be supersonic ($M > 1$).
2. Yaw is not allowed.
3. The present geometry package does not allow solution of a body with wings.
4. If the Mach number, M_∞, is below 2 and at the same time the angle of attack is above 15°, erroneous answers can occur due to large radial spacings of the grid near the body.
5. The body radius cannot be multivalued.
6. The program treats inviscid flow only.

The use of the program is illustrated in reference 18 by explaining the input and output for a number of test cases.

REFERENCES

1. Kutler, P., Lomax, H., and Warming, R. F.: Computation of Space Shuttle Flow Fields Using Noncentered Finite-Difference Schemes. AIAA Journal, Vol. 11, Feb. 1973, pp. 196-204.

2. Kutler, P., Reinhardt, W. A., and Warming, R. F.: Multi-Shocked, Three-Dimensional Supersonic Flowfields with Real Gas Effects. AIAA Journal, Vol. 11, May 1973, pp. 657-664.

3. Woods, B. A.: The Supersonic Flow Past an Elliptic Cone. The Aeronautical Quarterly, Vol. 20, 1969, p. 382.

4. Schiff, L. B.: Computation of Supersonic Flow Fields About Bodies in Coning Motion Using a Shock-Capturing Finite-Difference Technique. AIAA Paper 72-27, San Diego, CA, 1972.

5. Kutler, P., Lomax, H., and Warming, R. F.: Computation of Space Shuttle Flow Fields Using Noncentered Finite-Difference Schemes. AIAA Paper 72-193, San Diego, CA, 1972.

6. Kutler, P., Rakich, J. V., and Mateer, G. G.: Application of Shock Capturing and Characteristics Methods to Shuttle Flow Fields. NASA TM X-2506, Vol. 1, 1972, p. 65.

7. Morretti, G., Grossman, B., and Marconi, F., Jr.: A Complete Numerical Technique for the Calculation of Three Dimensional Inviscid Supersonic Flows. AIAA Paper 72-192, San Diego, CA, 1972.

8. Dekker, T. J.: Constructive Aspects of the Fundamental Theorem of Algebra. Edited by B. Dejon and P. Henrici, Wiley, New York, 1969, p. 37.

9. Rakich, J. V. and Kutler, P.: Comparison of Characteristics and Shock Capturing Methods with Application to the Space Shuttle Vehicle. AIAA Paper 72-191, San Diego, CA, 1972.

10. Abbett, M. J.: Boundary Condition Computation Procedures for Inviscid Supersonic Steady Flow Field Calculations. NASA CR-114446, 1971.

11. Kentzer, C. P.: Discretization of Boundary Conditions on Moving Discontinuities. Proceeding of the 2nd International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, Vol. 8, Springer-Verlag, Berlin, 1971.

12. Hayes, W. D. and Probstein, R. F.: Hypersonic Flow Theory. 2nd ed., Academic Press, New York, 1966, p. 485.

13. Barnwell, R. W.: A Time-Dependent Method for Calculating Supersonic Angle-of-Attack Flow About Axisymmetric Blunt Bodies with Sharp Shoulders and Smooth Nonaxisymmetric Blunt Bodies. NASA TN D-6283, 1971.

14. Thomas, P. D., Vinokur, M., Bastianon, R., and Conti, R. J.: Numerical Solution for the Three-Dimensional Inviscid Supersonic Flow. AIAA Journal, Vol. 10, No. 7, July 1972, pp. 887-894.

15. Morretti, G. and Abbett, M.: A Time-Dependent Computational Method for Blunt Body Flows. AIAA Journal, Vol. 4, No. 12, Dec. 1966, pp. 2136-2141.

16. MacCormack, R. W.: The Effect of Viscosity in Hypervelocity Impact Cratering. AIAA Paper 69-354, Cincinnati, OH, 1969.

17. Kutler, P.: Application of Selected Finite Difference Techniques to the Solution of Conical Flow Problems. Ph.D. Thesis, 1969, Dept. of Aerospace Engineering, Iowa State Univ., Ames, IA.

18. Chausee, D. S. and McMillan, O. J.: A Supersonic Three-Dimensional Code for Flow Over Blunt Bodies - Program Documentation and Test Cases. NASA CR-3224, 1980.

1. Report No.
NASA CR-3223

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
A SUPERSONIC, THREE-DIMENSIONAL CODE FOR FLOW OVER BLUNT BODIES - USER'S MANUAL

5. Report Date
January 1980

6. Performing Organization Code
512/C

7. Author(s)
D. S. Chaussee and O. J. McMillan

8. Performing Organization Report No.
NEAR TR 189

9. Performing Organization Name and Address
Nielsen Engineering & Research, Inc.
510 Clyde Avenue
Mountain View, CA 94043

10. Work Unit No.

11. Contract or Grant No.
NAS1-15305

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546

13. Type of Report and Period Covered
Contractor Report
3/1/78 to 4/30/79

14. Sponsoring Agency Code

15. Supplementary Notes
Langley Technical Monitors: Wallace C. Sawyer and Charlie M. Jackson, Jr.
Topical Report

16. Abstract

The technical explanation is given of a computer code which may be used to calculate the steady, supersonic, three-dimensional, inviscid flow over blunt bodies. The theoretical and numerical formulation of the problem is given (shock-capturing, downstream marching), including exposition of the boundary and initial conditions. The overall flow logic of the program is described, and its usage, accuracy, and limitations are discussed.

17. Key Words (Suggested by Author(s))
calculation methods
blunt bodies
supersonic flow
inviscid flow

18. Distribution Statement
Unlimited - Unclassified
Subject Category 02

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of Pages
25

22. Price*
$4.00

*For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1980